

PEER-TO-PEER MUSIC PLAYER



The Chinese University of Hong Kong
Department of Computer Science & Engineering

Peer-to-peer Karaoke System

Group 5

Poon Bing Chun	1155063232
Li Ka Chiu David	1155049531
Wong Chi Wa	1155062568

1 CONTENIDO

2	Basic information	1
3	Workload Division	1
4	Operation Manual	2
5	Third-party Libraries Used	5
6	Project Schedule	6
7	Further development	6
7.1	Support of more audio & video formats	6
7.2	Music video	6
7.3	Equalizer	6
8	Acknowledgements	6

2 BASIC INFORMATION

✚ Languages used: C++, JSON (markup), QMake (make script) & Qt Stylesheets (styling)

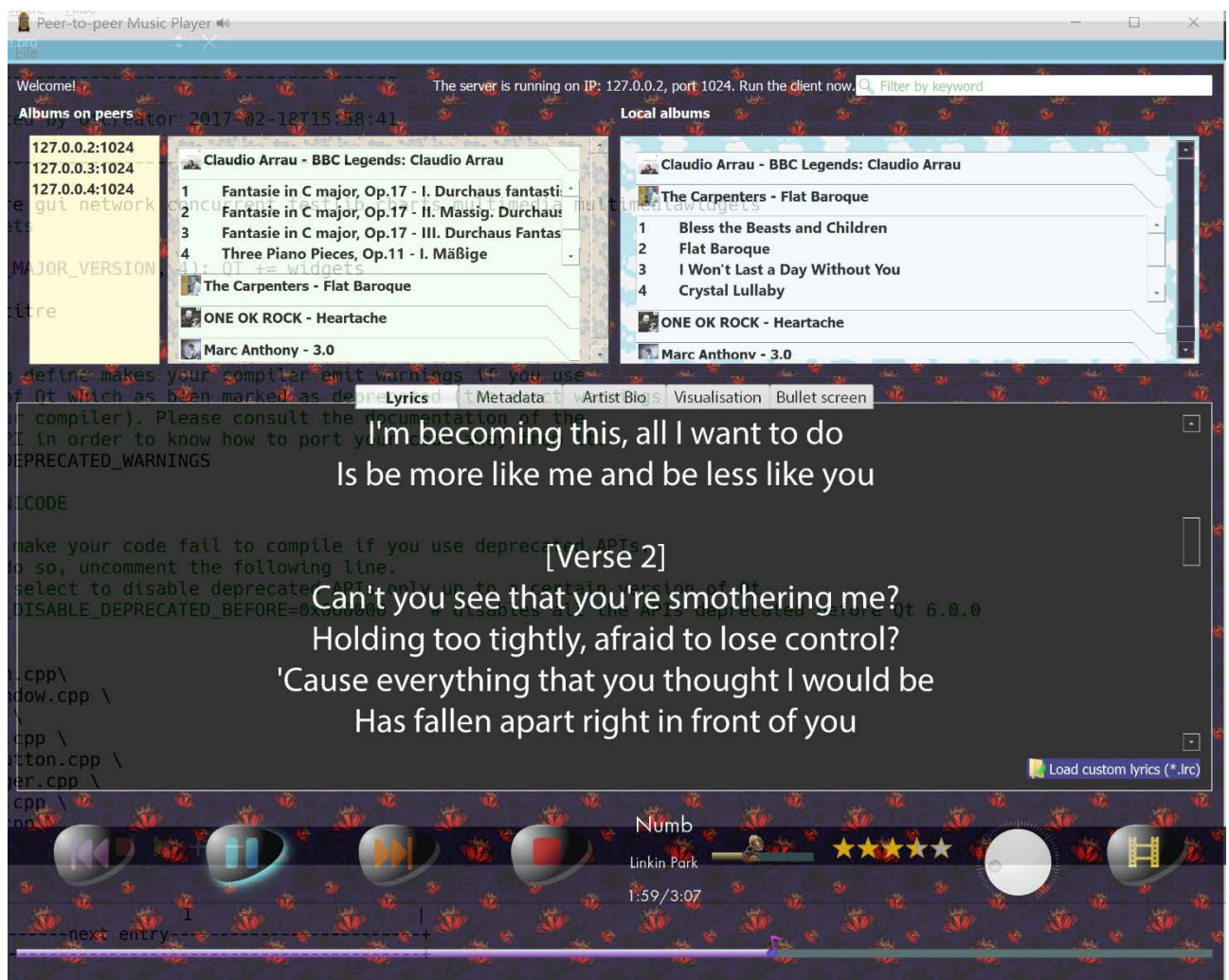
3 WORKLOAD DIVISION

Group member	Workload	Time spent
Poon Bing Chun	<ul style="list-style-type: none">✚ Graphical user interface✚ Automatic lyrics retrieval (Using HTTP requests, JSON, RegExp)✚ Multi-keyword search (RegExp)✚ Music catalogue parsing and transmission (JSON)✚ Volume control✚ Wave playback (only mutex, pause and stop)✚ Audio streaming✚ Multithreading✚ Project management (GitHub, Trello)✚ Code integration (Wave, video...)✚ Report (main author) Advanced features: <ul style="list-style-type: none">✚ Visualisation✚ Minimise to tray✚ Automatic album art, artist photo, album info, song info retrieval✚ Video playback visual effects	<ul style="list-style-type: none">✚ Please refer to the GitHub commit graph.
Li Ka Chiu David	<ul style="list-style-type: none">✚ Wave playback (main author) Advanced features: <ul style="list-style-type: none">✚ MP3 playback✚ Video playback (MP3, AVI, MPG...)✚ Report	<ul style="list-style-type: none">✚ Two months for the wave playback, and two weeks for MP3 and

		video playback.
Wong Chi Wa	<ul style="list-style-type: none"> Wave playback (to a lesser extent) Counselling Report (main author) 	Two to three weeks.
Wong Yee Lai	'Free-rider'	None

4 OPERATION MANUAL

- Splash screen and initial settings
- Main window



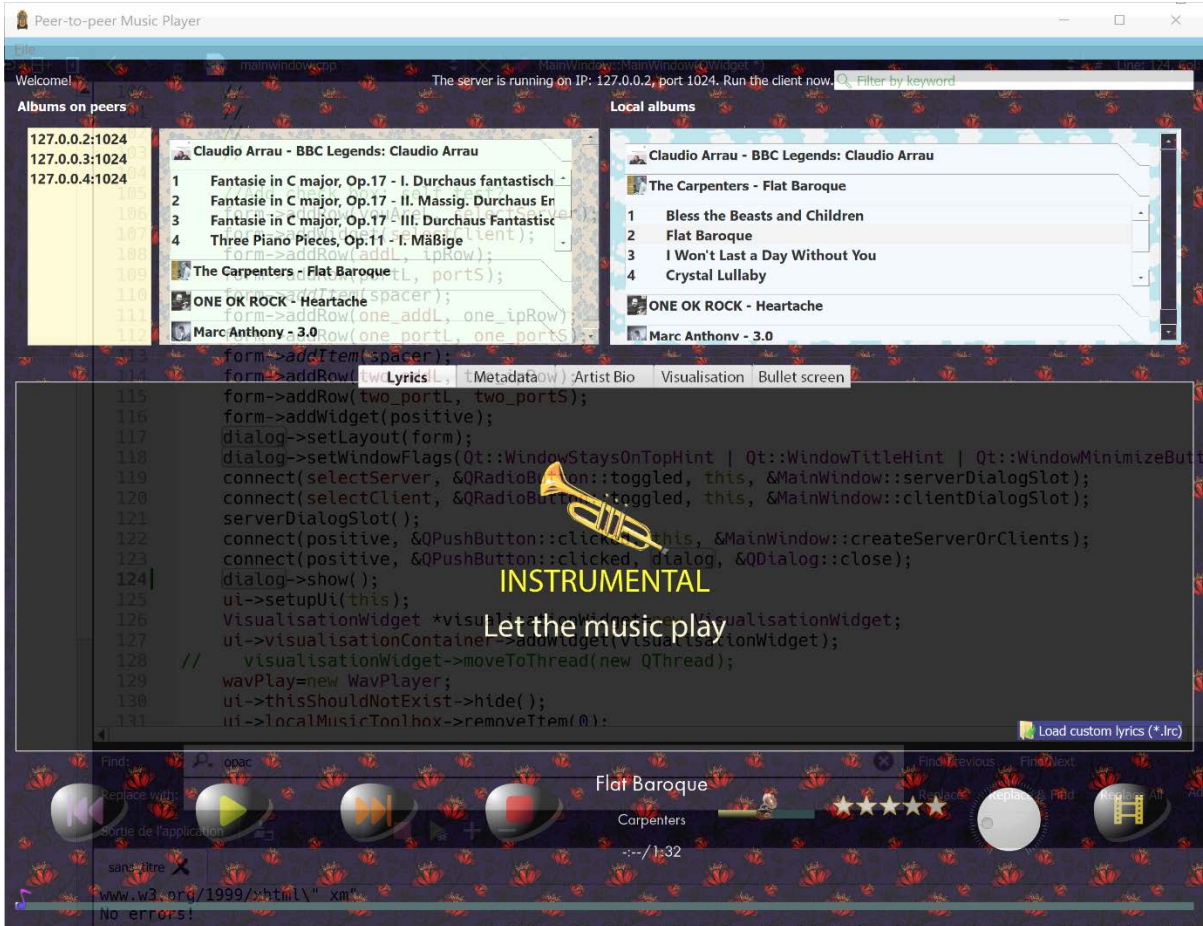
- Turn the knob control to adjust auto-scroll speed:



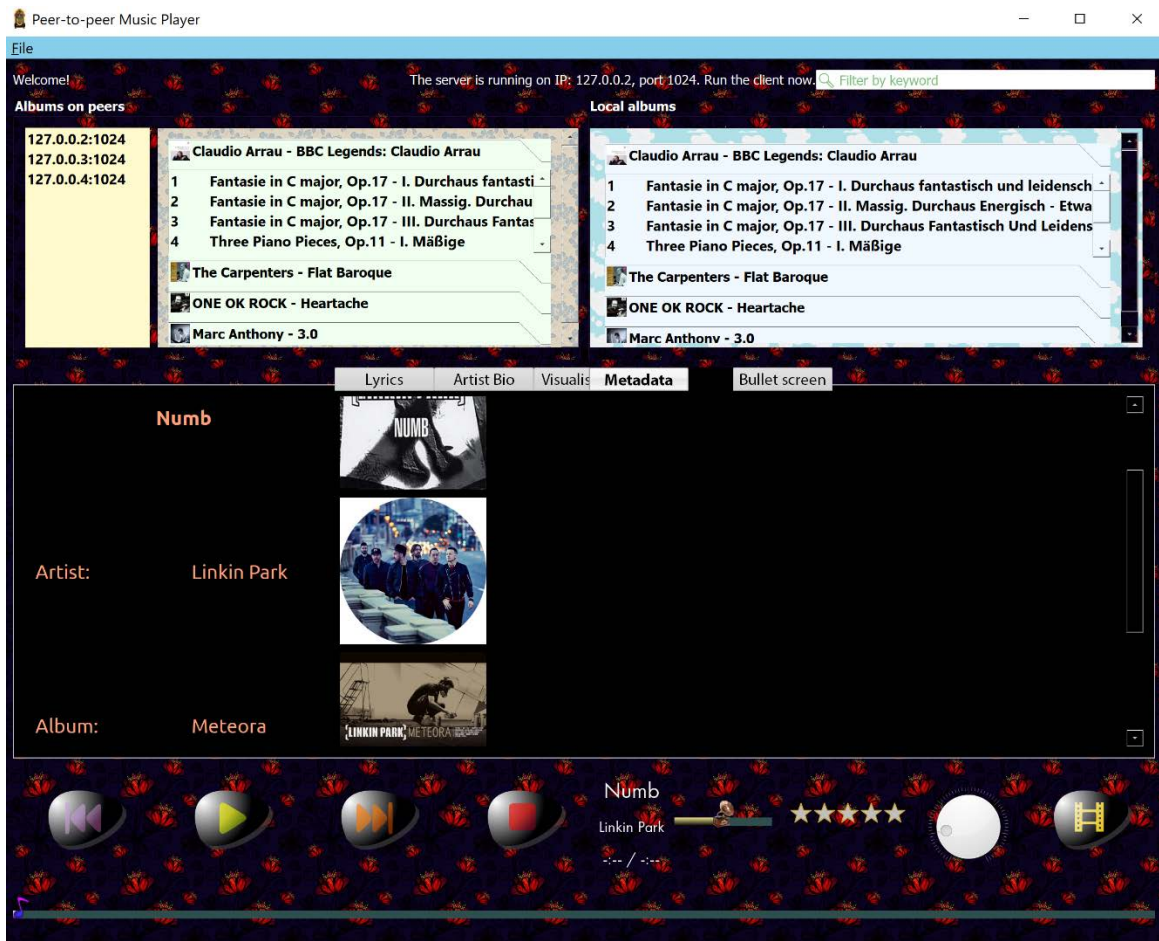
- Click this to launch the video player:



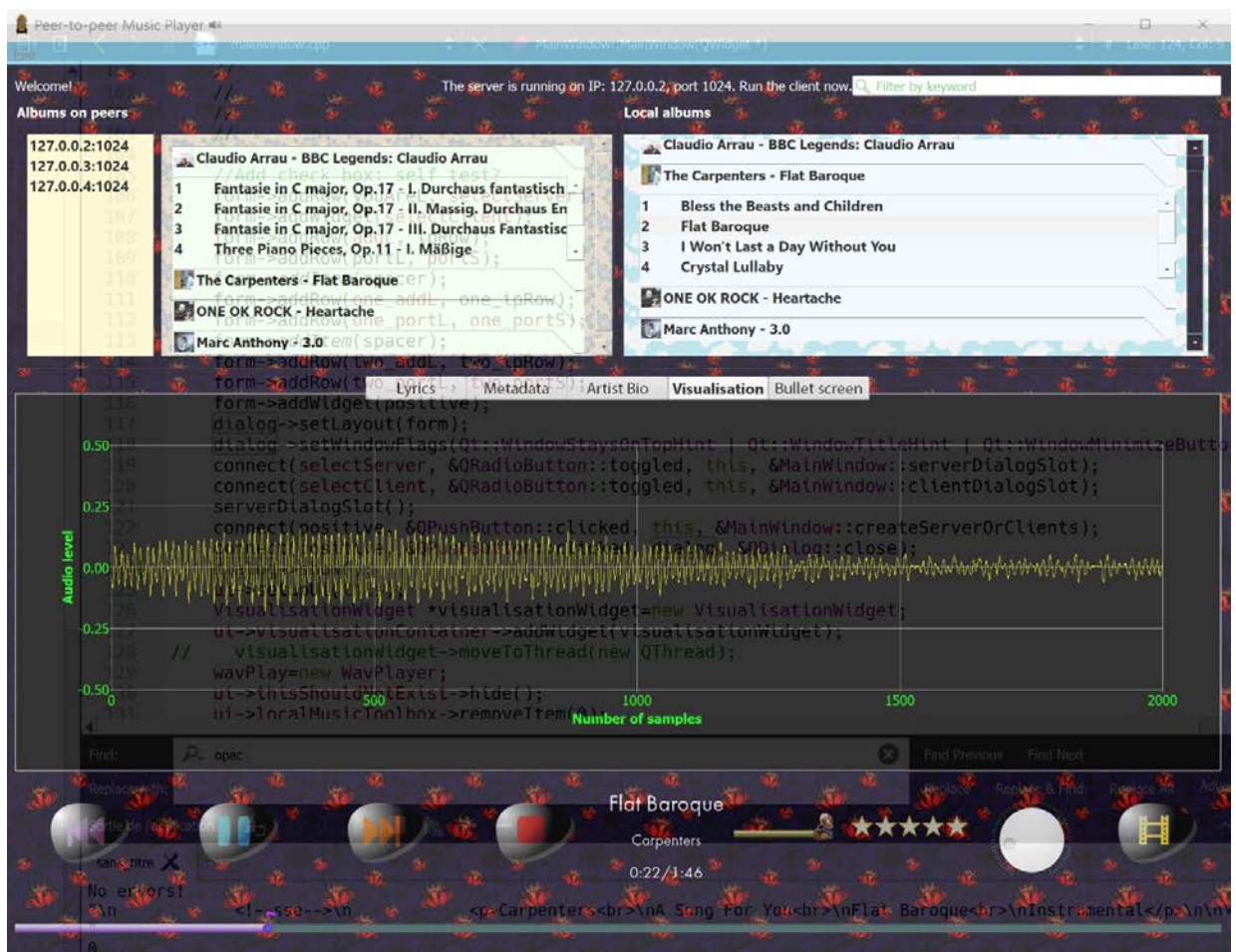
Instrumental song detected:



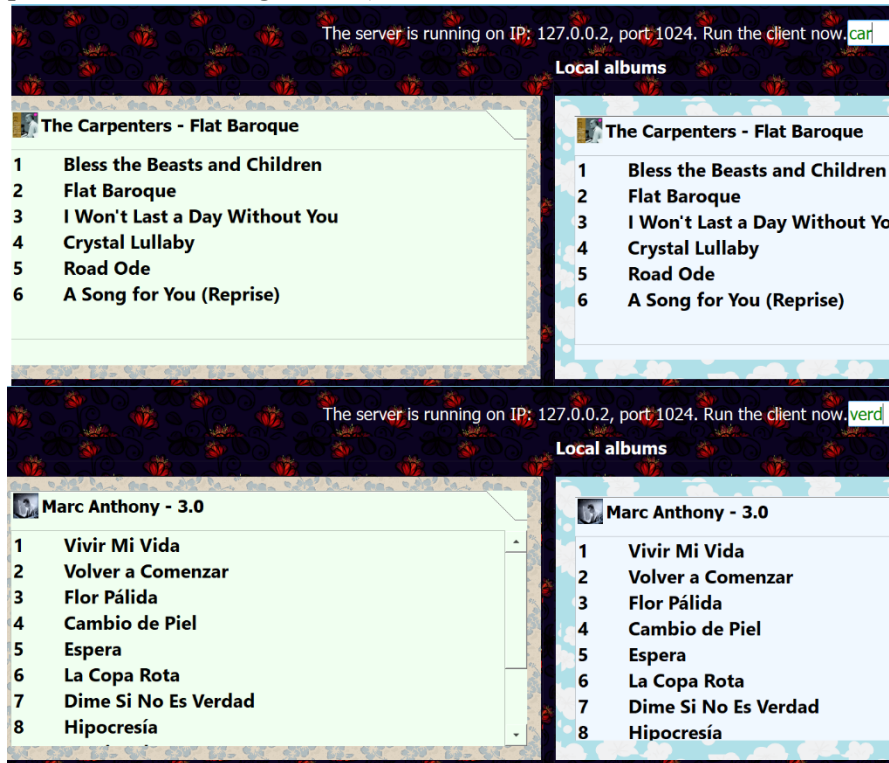
Album art, artist photo, album name retrieval. Also, draggable tabs.



Virtualisation (based on volume)

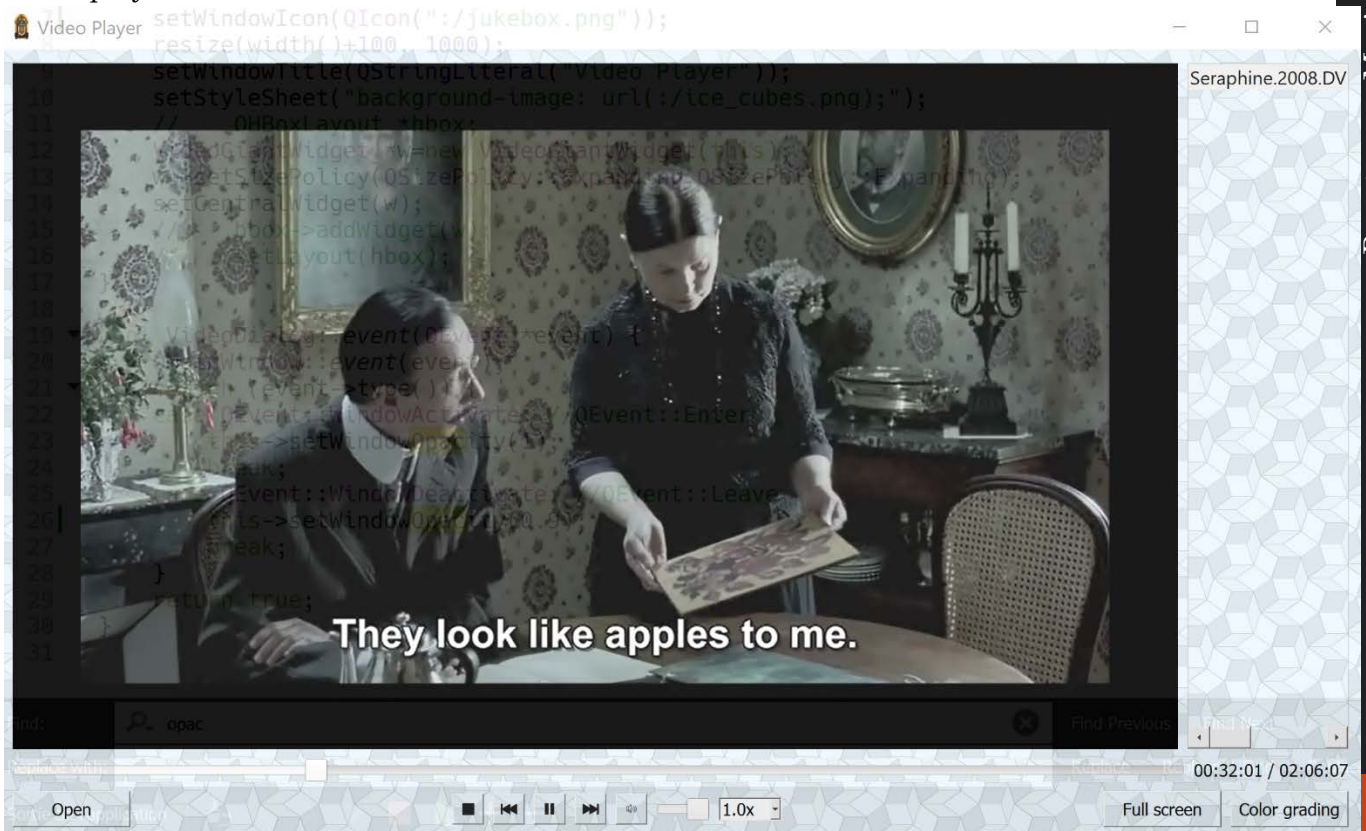


- Local music and remote music (multi-keyword search in action, it supports album, performer and song names)



- Draggable tabs

- Video player

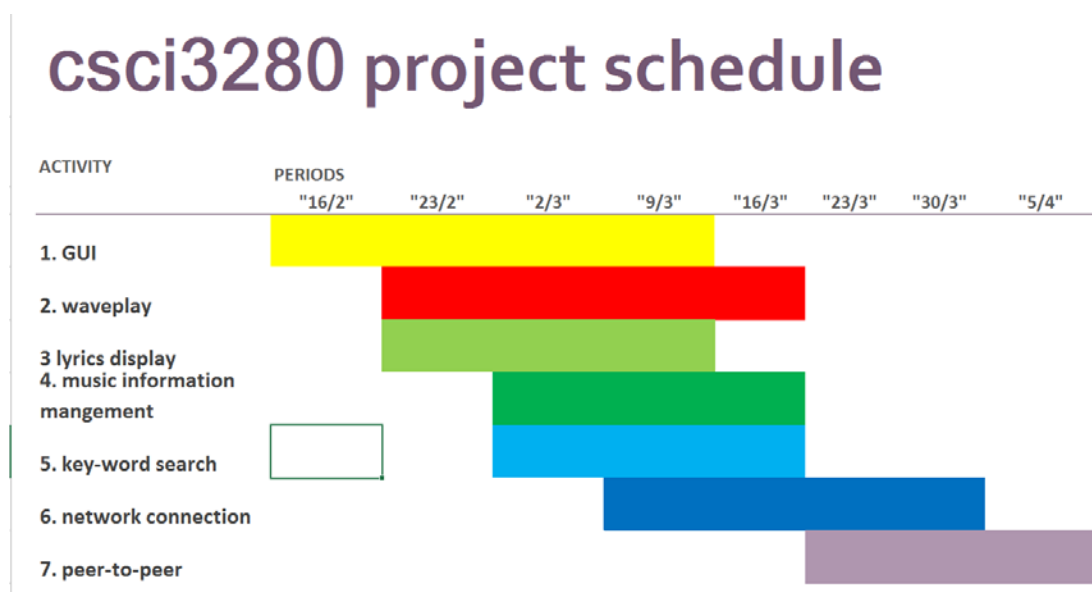


5 THIRD-PARTY LIBRARIES USED

- Qt (the modules are QtCore, QtGui, QtNetwork, QtConcurrent, QtCharts, QtMultimedia and QtMultimediaWidgets)

- ✚ Genius API (Album art, artist photo, album info and song info retrieval only. NOTE THAT lyrics retrieval is done using webpage retrieval and regular expression parsing, instead of this API.)

6 PROJECT SCHEDULE



7 FURTHER DEVELOPMENT

7.1 SUPPORT OF MORE AUDIO & VIDEO FORMATS

In the future, we aim to support a plethora of audio formats such as AIFF, ASF, FLAC, OGA, Speex and MPC; a well as video formats like OGV. Thus, users can enjoy different songs and videos without any inconvenience like format conversion.

7.2 MUSIC VIDEO

We also hope to support music video (MV) playback alongside songs, so that users can enjoy intense audio-visual entertainment.

7.3 EQUALIZER

We decide to add an equalizer to the player. It is used to enhance or reduce different frequency bands in the audio. The user can achieve different effects such as strong bass or large amount of high frequencies via the equalizer. It facilitates the flexibility and performance of the music player.

8 ACKNOWLEDGEMENTS

- ✚ Han Chu, our tutor
- ✚ GitHub for offering the Student Pack
- ✚ The Qt Company for offering a LGPLv3 license to use its API